DATE: 3/20/97

CONTROLLING OFFICE FOR THIS DOCUMENT IS:
SIMULATION AND TECHNOLOGY DIVISION
DIRECTORATE FOR TECHNICAL MISSION
US ARMY TEST AND EVALUATION COMMAND
ABERDEEN PROVING GROUND, MARYLAND 21005-5055

POC: DIRECTOR

DISTRIBUTION STATEMENT A: Public release

Computer Viruses:
Testing Will Never Be The Same

"Virus prompts partial EPA shutdown" was the headline in the November 11, 1996 edition of Federal Computer Week. The article below the headline described the infection of over 600 workstations with a resultant loss of information and a significant denial-of-service as personnel stopped all productive work to deal with the infection.

For almost a decade similar headlines have appeared throughout the Federal Government and in particular Department of Defense activities. Computer viruses in the development and test arena continue to destroy test data, to delay the completion of work, and to divert personnel resources away from mission-essential tasks.

Ironically software tools to address the viral phenomenon are almost as old as viruses themselves. Yet individual agencies either refuse to utilize such tools, or make questionable decisions in the selection of antiviral software. For those who wish to make an informed decision, it seems reasonable to ask what criteria may be important in acquiring and in using these tools.

U.S. Army White Sands Missile Range personnel have, over the last 8 years, conducted product evaluation tests of 43 antiviral identification and/or integrity checking programs. By conscious design the tests have included freeware, shareware, and commercial products for Windows 3.x, Windows 95, Windows NT, and the Macintosh operating system.

The following represents a modest attempt to summarize the results of those evaluations and to suggest selection criteria.

Effectiveness

Antiviral identification programs appear to detect what they say they can detect. While the suite of actual viruses and their variations is approximately 11,500, there have been only 6 products out of 36 which have not been able to reliably detect all the malicious code claimed for the Windows/DOS environment. Those programs with suspect reliability invariably failed to detect variations on known viral signatures. Since the test suite of malicious programs contains approximately 85 percent of the so-called most commonly reported infections, failure to detect variations on this category of infectors is obviously significant.

The seven Macintosh programs had perfect detection rates against those programs claimed to be detectable. In this instance the suite of malicious code included all known Macintosh viruses and variations with the exception of known HyperCard viruses. Admittedly, the Macintosh products have less variation in their advertised capabilities and far fewer malicious programs to combat. For example, one freeware and one shareware program claimed to detect all known Macintosh viruses with the exception of two HyperCard viruses. The shareware program also detected the three most prevalent trojan horses. The remaining commercial programs identified all known Macintosh viruses to include the same trojan horses.

While someone may argue that it is hazardous to extrapolate from the results against only 11,500 viruses, other sources – such as Internet discussion groups, reviews in computer trade publications, independent tests by the National Computer Security Association, by Virus Bulletin, and by the Computer Virus Centre - provide support for the extrapolation. Though there is the possibility that one author will reverse engineer another detection program to identify appropriate viral signature strings, the variation between the antiviral programs in terms of the total number of viruses claimed to be detected would seem to argue against the adoption of such a strategy. It is rather more plausible to conclude that the authors of these programs have constructed their products based upon what malicious code they have in hand.

The reality is that authors of detection programs have to obtain access to malicious code through a variety of means. With few exceptions the users of detection programs will not have such access. This raises the potential for a program author to claim by "emphatic assertion" that his or her product is the best. Many users and organizations may feel forced to select a product without some sort of confirmation as to its effectiveness. While there is currently no "Underwriters Laboratories" for antivirus software testing, Virus Bulletin, the National Computer Security Association, and the Computer Virus Centre in Hamburg routinely publish results on the effectiveness of programs to conduct detection and disinfection operations.

Numbers

While Windows/DOS programs in contrast to the Macintosh programs do indeed have significant differences in the number of malicious programs claimed to be detectable, these differences may be unimportant in most environments. It is difficult to quantify the actual number and type of infections which have occurred and which are occurring worldwide. There are many reasons for the failure to have trustworthy data. What statistics are available generally come from the authors or sources of the detection programs. These statistics reflect the reported infection rates of the respective product's user community. The surprise for many readers will be to learn that the statistics indicate as few as 25 viruses, and their variations continue to cause more than 90 percent of all reported infections.

If only a small number of viruses and their variations have such a major impact on infection rates, then one would be well advised to evaluate a product based upon what viruses it can actually detect. The total number of viruses detected becomes less of a criterion. There is an economic issue as well when one finds that a freeware program can detect the most common viruses and can be distributed throughout an organization without charge. Clearly the potential for an adversary or a disgruntled employee to implant a virus or malicious code which is not "common" is feasible. It is simply indeterminate at this time that such an "uncommon" attack has occurred.
Even where there is a variation, however slight, in the number of individual malicious programs identified by Macintosh products, the overwhelming majority of reported infections appear to revolve around the nVir, WDEF, MDEF, and most recently Word Macro viruses, including their variations or mutations. The respective authors or sources of Macintosh programs admittedly face the same problems as their Windows/DOS counterparts in collecting accurate statistical information on viral infections. Their difficulties are less when one considers that the number of known Macintosh viruses and trojan horses is insignificant when compared to those in the Windows/DOS environment.

Detection Strategy

There are different strategies for performing detection operations. The Windows/DOS products have varying formats and lengths in their search string composition. In certain programs a minimum of two search strings per individual virus or malicious program is the norm. There may be schemes to encrypt the search string. The intent in this instance is to make it more difficult for someone to modify known malicious code with the purpose of escaping detection. Another option, which perhaps implements the "security through obscurity principle" and which may or may not be a conscious strategy, is to simply refrain from discussing the composition of the product's search strings or detection techniques.

The Macintosh products, again because of the relatively small number of malicious programs, can afford to be more homogeneous and less combative among themselves as to overall detection strategies. There are also far fewer players in the Macintosh antiviral marketplace. For example, where there are dozens of antiviral detection programs for Windows/DOS systems, there may be only four to seven at the present time for the Macintosh. A single public domain program has literally taken over the Macintosh freeware market. There are perhaps two or three shareware programs which have withstood the technical judgment of the Macintosh community. Four commercial programs exist; two of which control the overwhelming marketshare. While there still may be variations in the actual search string composition or syntax, there exist more similarities than differences in overall detection strategy.

How much significance should one impart to those products which advertise dual search strings, or encrypted search strings? Are there instances in which someone has consciously reverse engineered an antiviral product to modify a known virus and avoid detection? There are examples in the Windows/DOS environment in which viruses have been created to escape detection or to make detection more difficult. Until the authors of those programs come forward and honestly discuss their intent and design philosophy, it would be premature to give a definite answer to the question of whether someone has consciously attempted to avoid a particular antiviral product. There is empirical data to conclude that known viruses have been modified and that these modifications have required formal revisions to detection programs.

There is confirmed evidence in the Macintosh community that an individual consciously constructed more than one virus to specifically avoid detection by freeware and commercial scanning programs. In this instance the limited sources for detection products became an advantage for the author in that the task was to avoid three or four programs, not dozens as might be the case in the Windows/DOS environment. The viral author was not modifying known viruses, but was using what he knew about detection programs to construct new viruses. Since both Windows/DOS and Macintosh products detect viruses and malicious code based upon a predefined search string or strings, the use of dual search strings or encrypted search strings has value to perhaps identify minor modifications or variations to known viruses, and to discourage widespread attempts to defeat a specific product. The dual search string approach may also introduce a measure of quality control, particularly where for whatever reason one of the search strings turns out to be an "exception" rather than a "norm" for the detection of a particular virus or malicious program.

It appears more logical that an individual would construct a "new" virus rather than reverse engineer a detection program to reinvigorate an "old" or known virus. Until there is more compelling evidence to suggest individuals are consistently targeting a specific detection product, I would not make dual search strings a mandatory criterion for a detection product. The criterion should be that at least the product source discusses the program's detection strategy and techniques. If the source uses "security through obscurity" and fails to address the subject, then one may choose to pursue this issue.

The encryption of search strings has some obvious defensive advantages to make reverse engineering more difficult. It also may help to eliminate type I or false positive alarms for viral signatures when one chooses to employ two or three different detection programs as part of an overall viral defense strategy.

Documentation

Product documentation varies significantly. One might assume that the more expensive detection programs would always provide the best documentation. This unfortunately does not hold true for either the Windows/DOS or the Macintosh programs tested. In the Windows/DOS environment an inexpensive shareware program actually provides the best tutorial on itself and on the malicious programs it claims to detect. In the Macintosh environment the freeware program takes the honors for its excellent overview of computer viruses and for its overall description of protection options and strategies. The more expensive products usually have larger manuals with better packaging. Size in itself does not guarantee either clarity or quality. There are also other factors which influence the documentation. As the cost of a product increases, one generally finds that detection is but one of several protection options found in the program. For example, options to disinfect and to protect against "new" or "unknown" viruses may be included with the detection capability. The integrated product will then require its documentation to address a wider subject matter.

Online documentation is another issue. The more expensive products generally provide far more assistance and advice to all categories of users. The one exception would again be the freeware Macintosh program.

The criteria in this instance would be to plan for the training of users and for the supplementation of any product's documentation. Even where both the printed and online documentation is of decent quality, users often do not read it. In those instances where organizations simply do not have the in-house resources to accomplish such training, one might incorporate this criteria into the actual product acquisition proposal and have the product source provide the training.

Detection Options

The operation of products has greater similarities than differences. The Windows/DOS products have common syntax commands and options. There are a set of baseline detection capabilities which program authors provide in response to simple logic rather than to any particular user demand. For example, the ability to generate an audit trail record of any detection operation is almost universal. The ability to receive an audio and/or visual alarm upon the detection of possible malicious code is standard. The ability to launch detection operations against multiple drives or to specifically direct operations against a directory of files is readily available.

The Macintosh architecture and operating system almost ensures that the "look-and-feel" of products reinforces the similarities. The menus and button displays eliminate the need for syntax commands. Users can launch operations as they would any other Macintosh application. The setting of detection options is almost identical across the products examined to date.

Most programs will provide a user with more options than he or she will ever routinely employ. Unfortunately product documentation does not always describe all options for the user. Therefore, given the similarities in product operation, the criteria may be to request that product documentation accurately and completely describes the options.

Site Specific Requirements

Notwithstanding the discussion on similarities, there are significant differences in product design and vendor support. This means that users must clearly understand their operating environments and must complete a risk assessment to finalize acquisition criteria. What are some of these differences that might be important?

Do you need a product with network detection capabilities? While all the programs work fine on stand-alone systems, networks present some difficulties. Even when a product advertises its network capabilities, the fine print may specify only a specific vendor's network.

Do you need a product which is menu-driven, or will your users accept a product which requires a specific syntax command statement? While this is not a concern in the Macintosh environment, it is for Windows/DOS users. Menus are more "user-friendly" and hopefully simplify training requirements. Menu-driven capabilities are generally the domain of commercial programs, although third parties have released front-end interfaces for several shareware antiviral programs. This seemingly trivial criterion could result in a significant economic impact on an acquisition decision. Do you need a product which allows you to update it when new malicious programs are identified, or can you afford to pay for upgrade and subscription services? Paying for updates, even with site licensing arrangements, can be an expensive proposition over the long term. While it is hazardous to predict the future of malicious software and viruses in particular, is there anyone willing to predict the demise of this phenomenon in the next 2 to 3 years? If not, then a long term economic analysis would seem an appropriate response to determine your specific criteria.

Do you need technical support from the product source, or will you have sufficient personnel resources to support the product? There is wide divergence in the support capabilities and technical expertise of those who provide detection products. If you need support, it would seem advisable to be very specific on your requirements. For example, if a conflict develops between the detection product and some other application, will the product source attempt to resolve it? If you discover what appears to be a new malicious program, will the product source analyze the program and update its detection program in a reasonable period of time? Will the product source provide 24-hour day support, 7 days a week? How long will it be before a product source answers a request for assistance? Admittedly not every user will have these types of concerns. That is why it is important to know your operating environment and to have an approved risk assessment which supports any acquisition decision.

Integrity Checkers

Antiviral identification programs generally detect known malicious signatures. This requires that a malicious sample be obtained and analyzed, and a reliable search string chosen. Since some researchers have stated publicly that they receive 30-40 "new" viral samples each day, identification by search string will at some point reach a point of diminishing returns. Indeed, several experts already believe that we have reached that point!

Integrity checking programs adopt a different approach. While they may incorporate identification through search string detection, they offer additional mechanisms to detect "changes" to a system as well as to detect "suspicious activity." The shareware and commercial programs offer a variety of options to generate a "signature" of a system and to detect variances to that baseline. The best offer full-menu interfaces so as to simplify the procedure.

While it is admittedly more difficult to test an integrity checker, detection of change clearly is an effective tool for antiviral and anti-malicious program defense. The marketplace is not so "busy" with vendors making claims and counterclaims for their integrity checkers, so the opportunity to avoid the hype exists.

There are at least two caveats for caution. First, there is confirmed evidence to suggest that antiviral authors have specifically targeted the integrity checker of several commercial vendors. This trend will probably continue forcing vendors and users to adopt additional operational and technical controls. Second, integrity checkers by their very operation will generate type I or false positive alarms. Change in itself does not always imply that one has a virus or a malicious program. So those who employ an integrity checker should prepare for false alarms and should maintain a sufficient technical staff to address user concerns.

Conclusion

The criteria discussed represent an attempt to assist users in the acquisition and selection of detection products. There are other factors which may be of equal, if not greater, importance. For this reason it is paramount that you actually test any product before you buy it. Most product sources will give you the opportunity to have an evaluation copy. When you test, have users of different skill levels experiment with the program. If only the expert user participates in the evaluation, you may choose a product which only the expert can understand and use.

The National Institute of Standards and Technology, Computer Security Division, has issued Special Publication 800-5, "A Guide to the Selection of Anti-Virus Tools and Techniques," December 2, 1992, which examines in greater detail many of the criteria suggested here.

A final suggestion would be to select not one, but rather at least two products for your enterprise. The dual product approach can address a host of potential problems which might develop if you simply commit all your resources to a single program. With two products you have greater flexibility. For example, it may be essential to give every user the ability to detect malicious programs and viruses. On the other hand, it may be too expensive and unnecessary to provide every user with a disinfection capability. A risk assessment may conclude that disinfection tools can be stored at a central location and then called upon when required. It is not necessary that you acquire an equal number of copies of each program, because you can adjust the quantity based upon your stated criteria and upon your overall information system security objectives.

For more information contact
Chris McDonald
Commercial (505) 678-3233
DSN 258-3233